

Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of the claims:

1. (currently amended) A method of creating a filesystem with transaction based functionality, comprising:
 - receiving an indicator to initiate a transaction for files stored in one or more portions of the filesystem;
 - duplicating the ~~{one or more portions of the}~~ filesystem within a pseudo-filesystem; and
 - creating a control text file that provides a textual filesystem interface and receives text-based commands to operate on the pseudo-filesystem;
 - processing the text-based commands written to the control text file; and
 - operating on ~~the one or more portions of the pseudo-filesystem within a~~ transaction according to the text-based commands.
2. (currently amended) The method of claim 1 wherein the duplicating is performed lazily in order to reduce processing impact on the filesystem.
3. (canceled)
4. (original) The method of claim 1 further comprising:
 - completing the transaction upon receipt of a text-based command associated with terminating the transaction.
5. (original) The method of claim 3 wherein the text-based commands include functional equivalent commands associated with terminating the transaction and selected from a set of commands for performing one of the following functions: delete directory, delete filesystem, and abort.
6. (currently amended) The method of claim 1 further comprising: updating the

filesystem with the updates performed on the pseudo-filesystem when the transaction has completed.

7. (original) The method of claim 6 wherein the updates are performed upon receipt of an indication to commit the transaction.

8. (original) The method of claim 1 further comprising:

creating a status text file that provides text-based status results from operations performed on the pseudo-filesystem.

9. (currently amended) The method of claim 1 wherein the indicator to initiate the transaction results from the creation of a directory within the a pseudo-filesystem.

10. (original) The method of claim 1 wherein the transaction ensures atomic updates to the filesystem in accordance with modifications made to the pseudo-filesystem and related files during the transaction.

11. (original) The method of claim 1 wherein a user assists in reconciliation of conflicts between updates in the pseudo-filesystems.

12. (currently amended) A method of interfacing with a filesystem comprising:

receiving a text-based command in a command file for operating on a pseudo-filesystem corresponding to the filesystem within a transaction;

determining whether one or more data dependencies would prevent the text-based command from being performed on the pseudo-filesystem; and

performing the text-based command to modify a file in the pseudo-filesystem; and potentially updating the pseudo-filesystem, the filesystem and one or more corresponding files associated with the pseudo-filesystem and filesystem respectively; and

updating the filesystem to include modifications performed to the file in the pseudo-filesystem; and

updating a status file associated with the pseudo-filesystem with a text-based status result for performing the text-based command and updates performed in the filesystem.

13. (currently amended) The method of claim 12 further comprising:

creating an entire copy of the filesystem;

mounting the entire copy of the filesystem under the pseudo-filesystem, wherein the text-based status results in the status file includes intermediate status result. [further comprising:

updating a status file associated with the pseudo-filesystem with text-based intermediate status results for performing the text-based command and updates performed in the system.]

14. (currently amended) The method of claim 12 further comprising:

creating a textual interface;

receiving the text-based command from a user into the textual interface, wherein the text-based status results in the status file includes final status results. [further comprising:

updating a status file associated with the pseudo-filesystem with text-based results indicating the final status associated with the command.]

15. (original) The method of claim 12 wherein receiving a text-based command includes functional equivalent commands selected from a set including: change root directory, select concurrency control type, select isolation level, commit transaction, and abort transaction.

16. (original) The method of claim 12 wherein determining the one or more data dependencies includes using optimistic concurrency control (OCC) to control pending read and write operations to the pseudo-filesystem, the filesystem and one or more corresponding files associated with the pseudo-filesystem and filesystem respectively.

17. (original) The method of claim 12 wherein determining the one or more data dependencies includes using lock-based concurrency control (LBCC) to control pending read and write operations to the pseudo-filesystem, the filesystem and one or more corresponding files associated with the pseudo-filesystem and filesystem respectively.

18. (original) The method of claim 12 wherein a user assists in reconciliation of conflicts between resources in the filesystem and pseudo-filesystems and files associated with these.

19. (currently amended) A computer program product for creating a filesystem with transaction based functionality, tangibly stored on a computer readable medium, comprising instructions operable to cause a programmable processor to:

- receive an indicator to initiate a transaction for files stored in one or more portions of the filesystem;
- duplicate ~~the one or more portions of~~ the filesystem within a pseudo-filesystem;
- ~~and~~
- create a control file that provides a textual filesystem interface and receives text-based commands to operate on the pseudo-filesystem;
- process the text-based commands written to the control file; and
- operate on ~~the one or more portions of the pseudo-filesystem within a transaction~~ according to the text-based commands.

20. (currently amended) A computer program product for interfacing with a filesystem, tangibly stored on a computer readable medium, comprising instructions operable to cause a programmable processor to:

- receive a text-based command in a command file for operating on a pseudo-filesystem ~~that is a copy of corresponding to the filesystem within a transaction;~~
- determine whether one or more data dependencies would prevent the text-based command from being performed on the pseudo-filesystem; ~~and~~

perform the text-based command ~~to update and potentially updating~~ the pseudo-filesystem; ~~the filesystem and one or more corresponding files associated with the pseudo-filesystem and filesystem respectively; and~~

update the filesystem to include modifications made to the pseudo-filesystem according to the text-based command; and

update a status file associated with the pseudo-filesystem with a text-based status result for performing the text-based command and updates performed in the filesystem.

21. (currently amended) An apparatus that creates a filesystem with transaction based functionality comprising:

a processor;

a memory having instructions capable of being executed on the processor that receive an indicator to initiate a transaction for files stored in one or more portions of the filesystem, ~~duplicate the one or more portions of the filesystem within a pseudo-filesystem, file processing text-based commands written to the control file; and operating on the one or more portions of the pseudo-filesystem within a transaction according to the text-based commands; stem within a pseudo-filesystem;~~ create a control file that provides a textual filesystem interface for receiving receives text-based commands to operate on the pseudo-filesystem, process the text-based commands written to the control file and operate on the one or more portions of the pseudo-filesystem within a transaction according to the text-based commands.

22. (currently amended) An apparatus that interfaces with a filesystem, comprising:

a processor;

a memory having instructions capable of being executed on the processor that receive a text-based command in a command file for operating on a pseudo-filesystem corresponding to the filesystem within a transaction, determine whether one or more data dependencies would prevent the text-based command from being performed on the pseudo-filesystem, and perform the text-based command ~~to update and potentially updating~~ the pseudo-filesystem, update the filesystem to include changes performed to the pseudo-filesystem according to the text-based command, ~~the filesystem and one or~~

~~more corresponding files associated with the pseudo filesystem and filesystem respectively;~~ and update a status file associated with the pseudo filesystem with a text-based status result for performing the text-based command and updates performed in the filesystem.

23. (currently amended) An apparatus for creating a filesystem with transaction based functionality, comprising:

means for receiving an indicator to initiate a transaction for files stored in one or more portions of the filesystem;

~~means for duplicating the one or more portions of the filesystem within a pseudo filesystem; and~~

means for creating a control file that provides a textual filesystem interface and receives text-based commands to operate on the pseudo filesystem;

means for processing the text-based commands written to the control file; and

means for operating on ~~the one or more portions of the pseudo filesystem within a~~ transaction according to the text-based commands.

24. (currently amended) An apparatus for interfacing with a filesystem, comprising:

means for receiving a text-based command in a command file for operating on a pseudo filesystem corresponding to the filesystem within a transaction;

means for determining whether one or more data dependencies would prevent the text-based command from being performed on the pseudo filesystem; ~~and~~

~~means for performing the text-based command; and potentially updating the pseudo filesystem, the filesystem and one or more corresponding files associated with the pseudo filesystem and filesystem respectively; and~~

means for updating the filesystem to include modifications performed to files and directories in the pseudo filesystem; and

means for updating a status file associated with the pseudo filesystem with a text-based status result for performing the text-based command and updates performed in the filesystem.